Laboratorul 2

Windows Registry:

Windows Registry reprezinta o structura ierarhica in care sunt salvate diferite informatii legate de configuratia sistemului de operare precum si despre diferite programare care opteaza sa pastreze informatii in aceasta locatie.

Inainte de Windows Registry, fiecare aplicatie isi salva configuratia intr-un fisier de initializare, deobicei cu extensia INI. Acest fisier, era deobicei impartit in mai multe sectiuni, iar fiecare sectiune avea mai multe valori.

De ex, un fisier de configuratie pentru un joc, ar putea arata in felul urmator

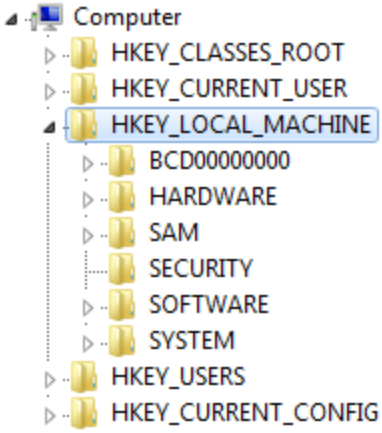
```
[Screen]
Resolution=”1024x768”
ColorDepth=32
[Player]
Name=”Player1”
Color=”red”
Character=1
```

Elementele intre paranteze patrate sunt sectiunile, iar elementele caror le sunt atribuite date, sunt valori.
Scopul pentru care a fost introdus Windows Registry, a fost sa centralizeze toate aceste informatii pentru toate programele instalate de pe sistem, intr-un singur loc.

Se poate lucra cu Windows Registry, folosind **regedit.exe** din Windows.

Structura

Structura la Windows Registry, este foarte asemanatoare cu sistemului de fisiere din windows. Daca radacina pentru sistemul de fisiere este o litera (de obicei atribuita unei partitii de pe disk sau unui dispozitiv extern , de ex: CD-ROM), iar informatiile sunt tinute in fisiere ce sunt organizate in directoare, in Windows Registry, locul partiilor sunt luate de niste structuri numite HIVE-uri, locul directoarelor de Chei, iar locul fisierelor de Valori.



In Stanga, este o imagine din regedit.

Structurile imediat de sub Computer, sunt Hive-uri (de ex, HKEY_CLASSES_ROOT), iar structurile de sub hive-uri sunt chei (de ex, Software)

Sunt 7 hive-uri in total (in paranteza e prescurtarea)

- HKEY_LOCAL_MACHINE (HKLM)
- HKEY_CURRENT_CONFIG (HKCC)
- HKEY_CLASSES_ROOT (HKCR)
- HKEY_CURRENT_USER (HKCU)
- HKEY_USERS (HKU)
- HKEY_PERFORMANCE_DATA (HKPD)

HKEY_LOCAL_MACHINE:

Tine informatii specifice calculatorului curent, cum ar fi:

- Informatii de login, useri, grupuri, din ce domeniu face parte sistemul (in cheia SAM)
- Informatii despre politicile de securitate (in cheia Security)
- Informatii despre dispozitivele isntalate pe sistem si mai multe configuratii (inclusiv una de back-up) (in cheia System)
- Informatii despre programele instalate pe sistem si configuratii pentru programe. Configuratiile salvate aici, sunt cele default. Aici sunt salvate atunci cand utilizatorul este intrebat daca vrea sa instaleze un program pentru utilizatorul curent, sau pentru toti utilziatorii, iar el selecteaza toti utilizatorii. (in cheia Software)
    - O subcheie foarte importanta din cheia HKLM\Software, este cheia Wow6432Node. In aceasta cheie sunt salvate aceleasi informatii ca si in software, doar ca aceasta cheie este folosita de aplicatiile care sunt compilate pentru 32 de biti, dar ruleaza pe un sistem de 64 de biti. Mai exact, Daca o aplicatie pe 32 de biti va dori sa scrie in cheia HKLM\Software\test, atunci intr-un mod transparent, ea va scrie de fapt in HKLM\Software\Wow6432Node\test

HKEY_CURRENT_CONFIG :

Tine informatii despre dispozitivele instalate pe sistem. Este de fapt o legatura la una din cheile din HKLM. (HKEY_LOCAL_MACHINE\System\CurrentControlSet\Hardware Profiles\Current)

HKEY_CLASSES_ROOT:

Contine informatii despre asocierea fisierelor cu aplicatii. (De exemplu, cu ce program sa fie deschise fisierele .docx sau .jpg).

HKEY_USERS:

Contine informatii specifice fiecarui utilizator. Cate o cheie pentru fieacre user si pentru fiecare grup. Informatile contin datede la imaginea folosita ca wallpaper si ce programe vor porni cand se autentifica utilizatorul pana la programele instalate doar pentru utilizatorul respectiv).

HKEY_CURRENT_USER:

Este de fapt o legatura la una din subcheile din HKEY_USERS si contine setarile pentru utilizatorul curent.

HKEY_PERFORMANCE_DATA:

Aceasta cheie nu este salvata nicaieri. Este doar o modaltiate oferita de sistemul de operare pentru a afla diferite informatii legate de performanta sistemului (atat informatii din kernel cat si user-mode).

Asa cum cheile sunt asemanatoare directoarelor de pe disk, valorile sunt asemanatoare fisierelor. Asadar, fiecare valoare, la fel ca si fisierele, poate fi de mai multe tipuri. Cele mai des folosite tipuri sunt urmatoarele:

- REG_SZ: tine un sir de caractere terminat prin 0
- REG_BINARY: orice tip de data, tinuta ca un sir de bytes
- REG_DWORD: un numar pe 32-biti fara semn (LITTLE-ENDIAN)
- REG_QWORD: un numar pe 64-biti fara semn (LITTLE-ENDIAN)
- REG_EXPAND_SZ: tot un string, doar ca deobicei contine diferite variabile Care vor fi inlocuite cu alte date. De exemplu, poate contine %system% care va fi inlocuit cu C:\Windows\system32
- REG_MULTI_SZ: O lista ordonata de siruri de caractere

Lucru cu Windows Registry din Windows API

Spre deosebire de functiile de lucru cu fisiere, functiile de lucru cu registry intorc ERROR_SUCCESS (definit ca 0) in caz de succes sau un cod de eroare in caz ca functia esueaza.

Lucrul cu cheile din registry este la fel ca orice obiect din Windows. Intai se obtine un handle pentru cheie (moment in care se verifica si drepturile de acces), iar apoi se apeleaza diferite functii pentru a modifica elemente din cheie.

Lucru cu chei

Pentru crearea unei chei, se foloseste functia RegCreateKeyEx. Daca cheia exista, atunci o va deschide, altfel, o va crea

```
LONG WINAPI RegCreateKeyEx(
  _In_        HKEY hKey,
  _In_        LPCTSTR lpSubKey,
  _Reserved_  DWORD Reserved,
  _In_opt_    LPTSTR lpClass,
  _In_        DWORD dwOptions,
  _In_        REGSAM samDesired,
  _In_opt_    LPSECURITY_ATTRIBUTES lpSecurityAttributes,
  _Out_       PHKEY phkResult,
  _Out_opt_   LPDWORD lpdwDisposition
);
```

hKey	Handle la unul din parintele cheii pe care vreti sa o creati sau o constanta care defineste unul din hive-uri(ex: HKEY_LOCAL_MACHINE)
lpSubKey	Numele pentru noua cheie, sau path-ul catre ea daca hKey nu este handle-ul la parintele imediat. Ex: Software\Test
Rserved	0
lpClass	0
dwOptions	Setari specifice cheii. Cel mai des sunt folosite REG_OPTION_NON_VOLATILE si REG_OPTION_VOLATILE care specifica daca sa fie salvata sau nu cheia pe disk la reboot
samDesired	Drepturile specifice operatiilor care se doresc a fi facute pe cheie. Cel mai des, KEY_WRITE
lpSecurityAttributes	drepturi de securitate, in caz ca e nevoei
phkResult	Aici va fi handle-ul rezultat, in caz ca functia reuseste
lpdwDisposition	Specifica daca cheia exista si a deschis-o (REG_OPENED_EXISTING_KEY)sau a creat o noua cheie REG_CREATED_NEW_KEY

O functie pentru deschiderea unei chei, este RegOpenKeyEx.

```
LONG WINAPI RegOpenKeyEx(
  _In_      HKEY hKey,
  _In_opt_  LPCTSTR lpSubKey,
  _In_      DWORD ulOptions,
  _In_      REGSAM samDesired,
  _Out_     PHKEY phkResult
```

```
);
```

Tinand cont ca parametrii din aceasta functie se gasesc si printre parametrii de la RegCreateKeyEx, nu are sens sa mai fie explicati (au aceiasi utilizare)

Pentru **stergea** unei chei, se poate folosi functia RegDeleteKey sau, daca se doreste ca o aplicatie pe 32 de biti sa poata sterge o cheie specifica sistemului pe 64 de biti (care nu este in Wow6432Node) si invers, se poate folosi functia RegDeleteKeyEx.

```
LONG WINAPI RegDeleteKeyEx(
  _In_        HKEY hKey,
  _In_        LPCTSTR lpSubKey,
  _In_        REGSAM samDesired,
  _Reserved_  DWORD Reserved
);
```

hKey	Handle la unul din parintele cheii
lpSubkey	Path-ul catre cheie care se doreste sa fie stearsa
samDesired	KEY_WOW64_32KEY daca se doreste sa stearga o cheie din arborele specific pe 32 de biti sau KEY_WOW64_64KEY daca se doreste stergerea din arborele pentru 64 de biti
Reserved	0

RegDeleteKey(Ex) va esua daca acea cheie ce se doreste a fi stearsa contine subchei. Pentru a sterge un arbore de chei, folositi functia RegDeleteTree.

Pentru **enumerarea** cheilor si a valorilor din registry, se folosesc mai multi pasi.
Pasii sunt urmatorii:

1. Se apeleaza RegQueryInfoKey pentru a afla cate subchei si valori sunt pentru cheia curenta. Functia ofera si alte date importante cum ar fi dimensiunea maxima a numelui unei chei sau a unei valori. Aceste date sunt utile pentru a aloca un buffer suficient de mare

   ```
   LONG WINAPI RegQueryInfoKey(
     _In_         HKEY hKey,
     _Out_opt_    LPTSTR lpClass,
     _Inout_opt_  LPDWORD lpcClass,
     _Reserved_   LPDWORD lpReserved,
     _Out_opt_    LPDWORD lpcSubKeys,
     _Out_opt_    LPDWORD lpcMaxSubKeyLen,
     _Out_opt_    LPDWORD lpcMaxClassLen,
     _Out_opt_    LPDWORD lpcValues,
     _Out_opt_    LPDWORD lpcMaxValueNameLen,
     _Out_opt_    LPDWORD lpcMaxValueLen,
     _Out_opt_    LPDWORD lpcbSecurityDescriptor,
     _Out_opt_    PFILETIME lpftLastWriteTime
   );
   ```

   Majoritatea parametrilor sunt optionali, doar o parte sunt folositi mai des:

lpcSubKeys	Aici va fi pus numarul de subchei
lpcMaxSubKeyLen	Aici va fi pus dimensiunea maxima a numelui unui subchei. Va fi folosit mai departe pentru a sti cati bytes trebuie alocati pentru a putea memora orice subcheie
lpcValues	Aici va fi pus numarul de valori
lpcMaxValueNameLen	Aici va fi pus dimensiunea maxima a numelui unei valori
lpcMaxValueLen	Aici va fi pus dimensiunea maxima pentru datele unei valori

2. Intr-o bucla care merge de la 0 pana la numarul de chei, se apeleaza functia RegEnumKeyEx. Functia va scrie in parametrii sai informatii despre subcheia cu indexul dwIndex.

```
LONG WINAPI RegEnumKeyEx(
  _In_         HKEY hKey,
  _In_         DWORD dwIndex,
  _Out_        LPTSTR lpName,
  _Inout_      LPDWORD lpcName,
  _Reserved_   LPDWORD lpReserved,
  _Inout_      LPTSTR lpClass,
  _Inout_opt_  LPDWORD lpcClass,
  _Out_opt_    PFILETIME lpftLastWriteTime
);
```

hKey	Handle la cheia pentru care se doreste enumerare de subchei
dwIndex	Index-ul cheii pentru care se doreste informatii (deobicei contorul din bucla)
lpName	Aici vor fi puse numele cheii. Ideal ar fi ca bufferul dat aici ca parametru sa fie alocat in functie de rezultatul de la RegQueryInfoKey
lpcName	Aici va fi pus cati bytes au fost copiati in bufferul lpcName. Inainte de a apela functia, trebuie ca lpcName sa fie dimensiunea alocata a lui lpName

Restul parametrilor pot fi 0

Asemanator, se apeleaza si functia RegEnumValue. Tinand cont ca o valoare are un tip si niste date, vor fi parametrii specifici acestor informatii.

Lucru cu valori

Pentru a crea o valoare sau pentru a modifica datele unei valori, se foloseste functia RegSetValueEx

```
LONG WINAPI RegSetValueEx(
  _In_        HKEY hKey,
  _In_opt_    LPCTSTR lpValueName,
  _Reserved_  DWORD Reserved,
  _In_        DWORD dwType,
  _In_        const BYTE *lpData,
  _In_        DWORD cbData
);
```

hKey	Handle la cheie carei apartine valoarea
lpValueName	Numele valorii
Reserved	0
dwType	Tipul valorii: vezi constantele pe MSDN (REG_DWORD, REG_BINARY, REG_SZ, etc.)
lpData	Buffer care contine datele ce trebuie scrise in valaore
cbData	Dimensiunea bufferului lpData

Pentru a citi datele dintr-o valoare, se foloseste functia RegQueryValueEx

```
LONG WINAPI RegQueryValueEx(
  _In_         HKEY hKey,
  _In_opt_     LPCTSTR lpValueName,
  _Reserved_   LPDWORD lpReserved,
  _Out_opt_    LPDWORD lpType,
  _Out_opt_    LPBYTE lpData,
  _Inout_opt_  LPDWORD lpcbData
);
```

hKey	Handle la cheia carei apartine valoarea
lpValueName	Numele valorii
lpReserved	0
lpType	Aici va fi pus tipul valorii (REG_DWORD, REG_BINARY, REG_SZ, etc.)
lpData	Aici vor fi puse datele din valoare
cpData	Aici trebuie dat dimensiunea buffer-ului lpData, iar daca functia reuseste, aici va fi pus cati bytes au fost scrisi in lpData.

Pentru a inchide handle=ul catre o cheie, se foloseste functia RegCloseKey

Alte functii utile se gasesc la adresa:

https://msdn.microsoft.com/en-us/library/windows/desktop/ms724875(v=vs.85).aspx